Terminal and Network Quality of Service

Debargha Mukherjee, *Member, IEEE,* Eric Delfosse, Jae-Gon Kim, *Member, IEEE,* and Yong Wang,

***Abstract*—In order to cater to diversity of terminals and networks, efficient and flexible adaptation of multimedia content in the delivery path to end consumers is required. To this end, it is necessary to associate the content with metadata that provides the relationship between feasible adaptation choices and various media characteristics obtained as a function of these choices. Further, adaptation is driven by specification of terminal, network, user preference or rights based constraints on media characteristics that are to be satisfied by the adaptation process. Using the metadata and the constraint specification, an adaptation engine can take an appropriate decision for adaptation, efficiently and flexibly. MPEG-21 Part 7 entitled Digital Item Adaptation standardizes among other things the metadata and constraint specifications that act as interfaces to the decision-taking component of an adaptation engine. This paper presents the concepts behind these tools in the standard, show universal methods based on pattern search to process the information in the tools to make decisions, and present some adaptation use cases where these tools can be used.**

***Index Terms*—MPEG-21, Digital Item Adaptation, terminal and network constraints, decision-taking, adaptation, transcoding, requantization, rate shaping, scalable bit-streams.**

I. INTRODUCTION

HETEROGENEOUS multimedia content delivery infrastructures and consumption devices present a huge obstacle in universal media access. Indeed, consumers use a growing variety of terminals to access multimedia content, over an equally diverse variety of networks with dynamically varying throughputs. To maximize consumer experience and ensure Quality of Service (QoS) commensurate with terminal and network capabilities and conditions, as well as user preferences, it is essential to adapt multimedia content in the delivery path to end consumers. Note here QoS is used loosely and does not correspond to network level guarantees.

Additionally, the set of rich media content and formats to be delivered is growing fast. This justifies a drive towards adaptation engines or modules thereof that use a universal processing model – which do not need frequent upgrades to support new formats and can even support proprietary ones.

Adaptation of various standardized formats has been extensively studied in recent years [1]-[9]. Invariably the focus of such work is adaptation efficiency, since full decoding followed by re-encoding with parameters so that the terminal and network constraints are met, is often infeasible from complexity and delay considerations. This includes rate adaptation and resolution conversion for Discrete Cosine Transform (DCT) coded images [1][2], rate adaptation and spatial and temporal resolution conversion for pre-encoded MPEG-1/2/4 videos [3][4][5], object based transcoding [6], and rate-distortion-complexity optimized transcoding [7].

In many of these cases, there is a compute intensive decision-taking involved for choosing the right set of parameters for adaptation that yields an adapted version of the content meeting terminal and network constraints. The adaptation efficiency can be greatly improved if this process could be simplified, in particular by providing some metadata that conveys pre-computed relationships between feasible adaptation parameters and media characteristics obtained by selecting them. This metadata is also the only means of providing information that cannot be directly obtained from a compressed bit-stream, such as distortion/fidelity measures with respect to the original uncompressed data. The decision-taking process then just uses the information in the metadata along with terminal and network constraints to make decisions, without requiring any information extraction through complex content manipulation. Furthermore, a universal processing model for the decision-taking process in an adaptation engine can be derived, so that descriptions and engines created by different parties can interoperate.

Digital Item Adaptation (DIA) [10][11] is Part 7 of the interoperable Multimedia Framework currently being developed in the ISO/IEC MPEG standardization committee as MPEG-21 [12][13], and aims to standardize various descriptions, called tools, on the Terminal and Network key element, including the metadata supporting decision-taking and the constraint specifications as required for QoS.

The rest of the paper is organized as follows. In Section II, the model for an adaptation engine is presented along with an introduction to various DIA components. Section III presents the decision-taking framework in detail. In Section IV, the optimization problem to be solved by a universal decision-taking process is described, along with some strategies for solving it. In Section V we show a variety of adaptation use cases involving various formats where the framework can be effectively employed to make adaptation decisions. Finally, conclusions and future directions are presented in Section VI.

Manuscript received April 30, 2004, revised June 30, 2004. Part of this work was funded by the IST-OZONE project (IST-2000-30026).

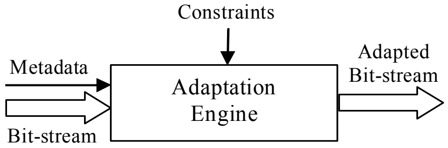
Debargha Mukherjee is with Hewlett Packard Laboratories, Palo Alto, CA, USA (phone: 650-236-8058; fax: 650-857-2951; e-mail: debargha@hpl.hp.com).

Eric Delfosse is with Interuniversity MicroElectronics Center (IMEC), Kapeldreef 75, B-3001 Leuven, Belgium (Phone: +32 16 28 10 68, Fax: +32 16 28 15 15, E-mail: delfosse@imec.be).

Jae-gon Kim is with ETRI, Korea (e-mail: jgkim@etri.re.kr).

Yong Wang is with the Electrical Engineering Department, University of Columbia University, NY, USA (e-mail: ywang@ee.columbia.edu).

(a) Top level adaptation engine



(b) Metadata enabled decision-making and bit-stream adaptation

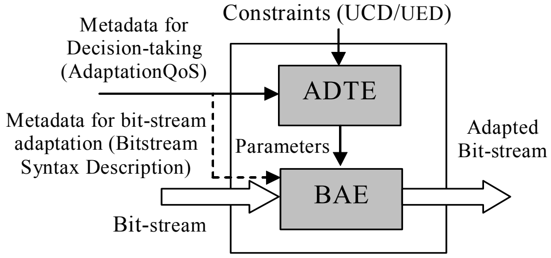


Fig. 1 Adaptation Engine model. ADTE – Adaptation Decision-Taking Engine; BAE – Bit-stream Adaptation Engine.

II. ADAPTATION ENGINE MODEL

The model of an adaptation engine envisaged in DIA is shown in Fig. 1. Fig. 1(a) is the top-level model of a metadata-enabled adaptation engine that absorbs various kinds of metadata to enable fast and efficient adaptation, for instance, based on terminal and network constraints. Fig. 1(b) shows a level more detailed model where the adaptation engine is decoupled into two functional modules: an Adaptation Decision-Taking Engine (ADTE) that absorbs metadata that aids decision-taking, as well as constraint specifications to make appropriate adaptation decisions; and a Bit-stream Adaptation Engine (BAE) that uses the decisions provided by the ADTE to perform the actual bit-stream adaptation.

An MPEG-21 DIA tool called the AdaptationQoS (AQoS) represents the metadata supporting decision-taking in the figure. Another DIA tool called the Universal Constraints Description (UCD) represents explicit constraint specifications. The adaptation constraints may also be specified implicitly by a variety of Usage Environment Descriptions (UED) providing network, terminal, user and environment characteristics and preferences that cover a major part of the standard. Examples of UED include display capabilities, audio/video capabilities of terminal, network characteristics, and so on. An ADTE directly using a UED input has to specifically understand the media-type context to make reasonable assumptions about how they translate to constraints to be applied to adapted content characteristics. Alternatively, only the data within a UED can be referenced within an explicit UCD, or the AQoS description, which enables a universal processing model for the ADTE.

The decisions made by the ADTE are next fed into the actual Bit-stream Adaptation Engine (BAE), which can be specific for a given format. This operation should be relatively simple given the decisions already made, so that the overall efficiency of adaptation is improved. For the special case of scalable bit-streams, such as JPEG2000 images [14] or fully scalable video proposed for standardization in MPEG-21 Part 13 [15][16][17], it has been shown [18][19] that by association of the content with additional metadata, both the ADTE as well as the BAE can have universal processing models, since the adaptation process simply involves removal of certain segments followed by update of certain fields as required for format compliance This leads to adaptation engines for scalable bit-streams that are fully format-independent. This additional metadata in DIA consists of the Bit-stream Syntax Description (BSD) [20][21] and a transformation stylesheet for the BSD. In the current paper, we focus on metadata that enables decision-taking and constraint specifications.

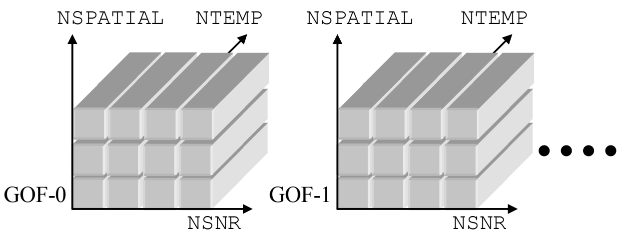
Note that while the metadata and constraint specifications are normative in DIA, the implementation of the ADTE and the BAE using them is non-normative. Further, DIA does not restrict in any way the delivery architecture within which a compliant engine is used is practice. The model is applicable in a variety of scenarios, irrespective of whether adaptation is conducted in the server/transmitter or in a gateway, whether the ADTE and BAE operations are distributed or occur at the same node, etc. A discussion of these architectural options and suitability for different delivery scenarios is beyond the scope of this paper. DIA also does not address network transport issues for either the metadata or the bit-stream, which is left entirely to the system implementation.

III. DECISION-TAKING FRAMEWORK

The purpose of the ADTE in an adaptation engine is to make an appropriate decision on how to adapt an input bit-stream, from among a set of available choices conveyed by the bit-stream description, based on specified input constraints. Furthermore, if the ADTE is to use a universal processing model it must not use any processing that is based on an understanding of the characteristics for specific media-types. This can only be accomplished if the decision-taking problem is expressed in the universal language of mathematics.

In MPEG-21 DIA, decision-taking is cast as a constrained optimization problem involving algebraic *variables* that represent adaptation parameters, media characteristics, usage environment inputs, or any combinations of the above. The solution, which yields the decision, can then be computed by a universal process, independent of what the variables represent. The framework further provides for the decision-taking functionality to be differentiated with respect to sequential logical segments corresponding to partitionings such as GOP, ROI, Tile, Frame etc., referred to as the *adaptation unit*. All variables are differentiated by the adaptation unit. For streamed content, the adaptation unit in many cases is also a unit of transmission comprising a bit-stream segment and corresponding metadata for decision-taking.

For example, consider a fully scalable video bit-stream [16] which contains simultaneous temporal, spatial and SNR scalability. It may also have color scalability, but we do not



`BITRATE` = *f*(`NTEMP`, `NSPATIAL`, `NSNR`) for each GOF

Fig. 2 Adaptation variables for fully scalable video

consider this case here. The video bit-stream is organized into multiple sequentially transmitted groups of frames (GOF), each typically containing 16 or 32 frames. The GOF constitutes the adaptation unit abstraction. Every GOF is coded jointly into several temporal, spatial and SNR layers. The number of layers included post adaptation for each GOF is represented by variables `NTEMP`, `NSPATIAL` and `NSNR`, forming a 3-dim logical hypercube structure for the decision space as shown in Fig. 2. Note that the exact algorithm used to generate the compressed layers is immaterial, since no matter what algorithm is used, this model will likely remain the same. Other variables such as `BITRATE` can be defined as a function of `NTEMP`, `NSPATIAL` and `NSNR`, for each GOF.

Denote the set of variables for the *n*th adaptation unit as vector $\mathbf{I}[n] = \{i_0[n], i_1[n], ..., i_{M-1}[n]\}$, $n = 0, 1, 2, \ldots$, where M is the number of variables. For each adaptation unit *n*, the optimization problem to be solved is given by:

$$\text{Maximize or Minimize } \{O_{n,j}(\mathbf{I}[n], \mathbf{H}[n])\}, j=0,1,...,J_n-1$$
$$\text{subject to: } L_{n,k}(\mathbf{I}[n], \mathbf{H}[n]) = \texttt{true}, \ k=0,1,\ldots,K_n-1$$

where $L_{n,k}(\mathbf{I}[n], \mathbf{H}[n])$, $k=0,1,\ldots,K_n-1$, are Boolean expressions called limit constraints, and $O_{n,j}(\mathbf{I}[n], \mathbf{H}[\text{n}])$, $j=0,1,\ldots,J_n-1$ are numeric expressions called optimization constraints. The number of optimization constraints (J_n) is arbitrary. If $J_n=0$, any solution in the *feasible* region – defined as the region of the solution space where the limit constraints are satisfied – is acceptable. The case $J_n=1$ is the most common and defines a single-criterion optimization problem which usually has a unique solution. The case $J_n>1$ defines a multi-criteria problem [22][23], where any Pareto optimal solution in the feasible region is accpetable.

Let $\mathbf{I}^*[\text{n}]$ represent a solution to this problem for the n^{th} adaptation unit. The vector $\mathbf{H}[\text{n}]$ in the expressions of $O_{n,j}$ and $L_{n,k}$ represents the history of all past decisions for adaptation units 0,1,…, *n*-1. In other words, $\mathbf{H}[\text{n}] = \{\mathbf{I}^*[0], \mathbf{I}^*[1], \ldots, \mathbf{I}^*[\text{n-1}]\}$. An ADTE makes decisions for the vectors $\mathbf{I}[n]$ sequentially for $n = 0,1,2,\ldots$ The dependency on history of past decisions is needed in certain cases, as in Section V.D.

The DIA tools, AdaptationQoS and UCD used in combination, support the above decision-taking mechanism. Variables are termed IOPins and are defined in the AdaptationQoS description. In cases involving multiple adaptation units, there is one IOPin defined in AdaptationQoS that indexes successive adaptation units, while other IOPins are functions of this IOPin. The AdaptationQoS description also conveys the known interdependencies between IOPins using various data types defined in the tool. These include look-up tables, numeric functions represented by an expression stack, or lists of values assumed for each adaptation choice termed utility functions. Note that the UCD or AdaptationQoS can still reference values from the UED, but the processing is driven by UCD or AdaptationQoS rules to ensure semantics-independent operation.

Note that the semantics of the IOPins are immaterial within the ADTE because they are simply regarded as mathematical variables to solve in a generic optimization problem. However, they are very much important at the provider and receiver ends or other nodes from where the AdaptationQoS or UCD originates. That is because, the UCD creator in many cases would not be expected to know the identifier of the IOPin (variable) defined in the provider side AdaptationQoS description, corresponding to a given semantics. In order to enable linking of the UCD to the right IOPins in AdaptationQoS, DIA creates a number of dictionaries termed *classification schemes* to standardize terms having pre-defined semantics for representing media characteristics, usage environment characteristics, and segment decompositions. The AdaptationQoS associates the IOPins it defines with terms that are the closest in semantics, while the UCD creator uses the same terms to specify the problem, rather than use identifiers of the IOPins directly. The ADTE simply performs a textual match of the classification scheme terms used in AdaptationQoS and UCD to know how the constraints specified in UCD using semantics terms map to IOPins.

IV. ADTE OPTIMIZATION

Generally speaking, an ADTE can have several inputs to it, comprising an AdaptationQoS, and several UCDs or UEDs from various sources. Based on these inputs, the ADTE needs to make appropriate adaptation decisions, by solving one or more constrained optimization problems [24]. We first discuss the single UCD case, and then present options to cover multiple UCDs originating from different sources.

A. Optimization problem involving free variables

The AdaptationQoS declares and defines several IOPins, some of which are independent, while others depend on other IOPins. Among the independent IOPins, some are assigned based on usage environment inputs either explicitly through the UCD or through data semantically referenced from a UED. Additionally, in cases involving multiple adaptation units there is one independent adaptation unit IOPin. The remaining independent IOPins, denoted $\mathbf{x}[\text{n}]$, comprise *N free* variables that need to be optimized. At the start of the optimization process, the ADTE performs simple analyses of the UCD and the AdaptationQoS to determine the free IOPins. Then it performs the optimization for each adaptation unit successively. A free IOPin can either be discrete, i.e. taking values from a finite discrete set, or continuous but bounded within two limits as provided in the IOPin definition in the AdaptationQoS. Denote $\mathbf{x}[\text{n}]=\{\mathbf{x}_d[\text{n}]' \ \mathbf{x}_c[\text{n}]'\}'$, where $\mathbf{x}_d[\text{n}]$ is the vector of N_d discrete free variables, and $\mathbf{x}_c[n]$ is the vector of N_c continuous free variables, and $N = N_c + N_d$. Also, note that the set of values for a discrete IOPin can either be unordered – corresponding to categorical variables, or ordered. Thus, the problem to be solved is a *mixed-variable multi-criteria optimization problem with general constraints* [22][24][34] defined over free IOPins for each adaptation unit. Fig. 3(a) shows the top level ADTE optimization flowchart.

For a universal ADTE, the generality of the optimization method used is of vital importance. Further, because the syntax of the UCD in MPEG-21 DIA allows representing arbitrarily non-linear functions for limit and optimization constraints, it is desirable to use methods that do not rely on derivative computations, but only black-box function computations as defined in the AdaptationQoS or the UCD.

B. Search/Optimization Strategies

For the problem as defined above, we consider first the discrete-only case, then the continuous-only case, and then extend to tackling the general mixed continuous-discrete case.

1) Discrete-only case:

In this case, all the free IOPins are discrete variables: $\mathbf{x}[n]= \mathbf{x}_d[n]$, $N = N_d$, $N_c = 0$. In commonly occurring adaptation decision-taking scenarios, it is sufficient to only search for the best parameters among a set of available discrete choices provided in the AdaptationQoS. Further, the number of free variables and the set of choices for each is small enough in the considered use cases, so that search by total enumeration in the discrete space of values the variables take, is not unfeasible. For this case, it is always possible to find a generic solution, irrespective of the nature of the functions and for any number of optimization constraints. Furthermore, the exhaustive search method is well suited for the nature of the UCD specification in the DIA standard, where the limit constraints are simply represented as Boolean functions that only return whether the constraint is satisfied at a given point. We outline the procedure for finding the entire solution set for arbitrary number of optimization constraints.

In multi-criteria optimization literature [22], a point $\mathbf{y}$ is said to *dominate* another point $\mathbf{z}$ with respect to a given set of optimization metrics, if one of the metrics evaluated at $\mathbf{y}$ is better than that evaluated at $\mathbf{z}$, and no other metric evaluated at $\mathbf{y}$ is worse than that evaluated at $\mathbf{z}$. A point $\mathbf{y}$ is said to be Pareto optimal, if there is no other feasible point that dominates $\mathbf{y}$. The goal is to find the set of all Pareto optimal solutions. Given N free variables for each adaptation unit, the ADTE starts with a null initialized list of solutions. Then it generates a N-tuple for each candidate solution, and evaluates the limit constraints to test feasibility. If feasible, the optimization metrics are evaluated, and the candidate is compared with the current running list of solutions to check for mutual domination. If the candidate is dominated by at least one other existing solution, it is discarded. If not, the candidate is added to the list, but existing solutions that are dominated by the candidate, if any, are discarded from the list. Once all N-tuples in the space of free variables have been processed, the list yields the Pareto optimal set of solutions. Any particular solution from this set can then be chosen as the final decision.

2) Continuous-only case:

In this case, all of the free IOPins are continuous variables: $\mathbf{x}[n]= \mathbf{x}_c[n]$, $N = N_c$, $N_d = 0$. We consider first the single-criterion optimization problem, and at the end generalize to the multi-criterion problem.

A family of optimization methods, known as *direct search* [25], that only rely on function computations at given points, is well suited for black box optimization problems. Early direct search methods [26][27] were simplex based. A subclass of direct search methods termed *pattern search* that evolved from [28], has recently been generalized under a common theory of Generalized Pattern Search (GPS) [25][29][30] with convergence properties stronger than simplex-based methods. These methods are especially suitable for the ADTE problem. GPS methods generate a sequence of iterates of the solution with non-increasing objective functions. At each iteration, a set of trial steps around an incumbent solution are searched, using directions independent of the optimization metric taken from an underlying lattice centered on the incumbent. A subset of the steps is required to form a positive spanning set to ensure convergence [30]. If a step is found where the objective function is reduced, the incumbent is moved to that point, while the lattice scale is either maintained the same or increased. Otherwise, the incumbent is maintained the same, but the lattice scale is reduced for the next iteration. The simplest embodiment of the general method is co-ordinate search or compass search, where the trial steps are in the positive and negative directions of each co-ordinate to make a total of $2N$ directions, at the current scale factor. This method is especially suitable for problems such as the ADTE, where the dimensionality N is not known a priori.

Direct search methods are basically for unconstrained problems, but can be adopted to handle constraints by converting a constrained problem into an unconstrained one by incorporating an exterior penalty term that penalizes the metrics if the limit constraints are violated based on the degree of violation. For a general optimization problem:

$$Minimize\,\{f(\mathbf{x}) : \mathbf{g}(\mathbf{x}) \le 0, \mathbf{h}(\mathbf{x}) = 0\}$$

where f is the objective function to be minimized with $\mathbf{g}$ and $\mathbf{h}$ being vectors $\{g_0, g_1, ..\}$ and $\{h_0, h_1, \ldots\}$ of inequality and equality constraint functions respectively, one form of the unconstrained function to be minimized is:

$$\phi(\mathbf{x},\mu) = f(\mathbf{x}) + \mu[\sum_i (Max[0, g_i(\mathbf{x})])^2 + \sum_i (h_i(\mathbf{x}))^2]$$

μ is a positive number called the penalty factor, that is usually iteratively increased to a very large value, since the optimal solution satisfying the constraints is achieved only as $\mu \to \infty$. An advantage of exterior penalty methods is that an initial feasible point is not needed. The difficulty however is that from the unstructured Boolean UCD specification in DIA, algebraic functions $g_i(.)$ and $h_i(.)$ as defined above, cannot be

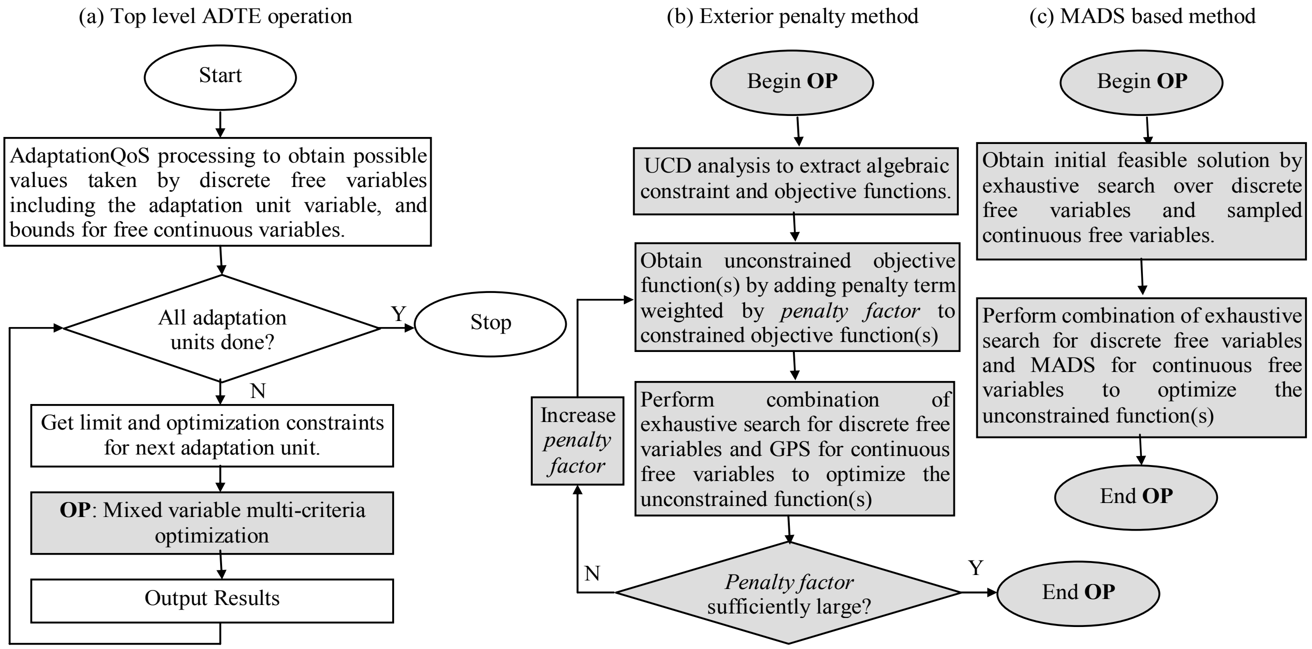


Fig. 3 Optimization strategies (a) Top-level ADTE operation; (b) Strategy based on exterior penalty method with GPS (requiring UCD-analysis); (c) Strategy based on MADS (cannot handle equality constraints)

directly obtained. It will be necessary to analyze and understand a UCD before an appropriate penalty term based on the degree of violation can be constructed. This processing step can be nontrivial.

In order to bypass the UCD analysis step, it would be convenient to use a method that simply uses Boolean black-box constraint functions that only return a yes/no answer indicating whether the constraints are satisfied at a trial point or not. In principle, GPS algorithms can be adapted to handle general constraints by an exact *barrier* approach that simply disregards trial steps that are infeasible, but Torczon argued in [25][31] that the search directions must be sufficiently dense and conform to constraint boundaries. A recent work by Audet and Dennis entitled Mesh Adaptive Direct Search (MADS) [32], specifically addresses this case, and is the only one of its kind so far. This method adds flexibility to GPS to search over a denser set of directions with a larger step size than the current mesh scale, to allow provable convergence for a large number of nonlinear constraints. However, as in all barrier methods, equality constraints cannot be directly handled. Nevertheless, for a generic UCD specification, this approach is the most promising.

In all direct search methods except those involving exterior penalty functions, there is a need to search for an initial feasible solution. Since the AdaptationQoS description provides the limits of the continuous variables, this search can be conducted by an exhaustive search in a coarsely sampled discretized space. One strategy is to use a solution to the discrete-only problem over the sampled space, as an initial feasible solution. If there are no feasible solutions in the sampled space, the sampling step can be reduced, as long as the number of points to search does not become prohibitively large. Otherwise, the ADTE just gives up reporting an error.

Finally, for multi-criteria problems pattern search methods are especially suitable, since only comparisons between points are needed. The criterion for comparing two points is simply changed to domination in the multi-criteria sense. Therefore, if any *one* Pareto optimal solution is desired, all the methods described above apply to multi-criteria problems as well.

3) Mixed discrete-continuous case:

In this case, some variables are continuous while the rest are discrete: $N_c \neq 0$, $N_d \neq 0$. This is the most general case for the ADTE problem. GPS/MADS based methods as described in the previous sub-section are equally applicable to these problems, with special handling for the discrete variables [33][34]. Trial steps in the mixed case should cover not only the pattern derived from a lattice defined over continuous variables as usual, but also all possible *neighbors* for the discrete variables. For the general case of categorical discrete variables, all values are neighbors of each other, and hence all possibilities need to be searched. Each step of the MADS iteration thus causes a change in either the discrete or the continuous variables.

The discussion in this sub-section is summarized by showing two viable strategies for handling the OP block in Fig. 3(a) when there are one or more continuous free variables involved – one based on exterior penalty functions shown in Fig. 3(b), and the other based on MADS shown in Fig. 3(c).

C. Handling multiple UCDs

When an ADTE receives multiple UCDs from various sources, each defining an independent optimization problem, it is necessary to combine them in some way. In such cases, it is reasonable to require that all the limit constraints combined from all the UCDs must be satisfied. That is, the feasible region for the solution should be the intersection of the feasible regions provided for the individual UCDs.

For the optimization constraints however, there are several options. The first is to combine all the optimization constraints from all UCDs into an integrated multi-criteria problem, and then to search for a Pareto optimal solution in the intersected feasible region. The second option is to prioritize the UCDs based on the source of the UCD, and only search for a Pareto optimal solution in the intersected feasible region for the optimization constraints in one of the UCDs, ignoring those from the others. For instance, the ADTE may decide to give a higher priority to the optimization constraints in the UCD sent from the provider side, over those sent from the consumer side, when both UCDs provide optimization constraints. The third option is to solve the problems independently for optimization constraints in different UCDs, and then to search for a solution in the intersection of the individual Pareto optimal solution sets. If the intersection is null, a prioritization mechanism may be used to pick a solution from the union of the individual solution sets. While this option is in many ways the best in terms of fulfilling the motivation of the individual UCDs, this is also the hardest to implement. In this case the ADTE should not only solve multiple problems, but also remember all possible Pareto optimal solutions for each. In contrast, the first two options require solving only a single problem, and do not even need storing all possible solutions.

Based on these considerations we advocate adopting either the first or the second approach or a combination thereof.

V. ADAPTATION USE CASES

In this section we describe a few use cases involving real bit-streams, specifying the decision-taking problem that aids adaptation, what metadata needs to be provided to enable decision-taking, some possible constraints that may be used to drive adaptation and how the decisions made are used for the actual bit-stream adaptation. All these cases use discrete optimization by exhaustive search in the decision space.

A. JPEG image adaptation

A classical adaptation problem for JPEG images (or MPEG intraframes) is that of requantization. The goal is to adapt a compressed JPEG image to a rate lower than the original. A problem associated with any viable method with JPEG is that there is no guarantee of the adapted rate achieved. Consequently multiple passes are required to avoid both violating the rate constraint or over-adaptation resulting in high distortion. This problem can be readily handled however by providing pre-computed information in the AdaptationQoS that maps possible parameter values for a specific BAE scheme to the rate and distortion achieved for a given image. The methodology below applies not only to JPEG images but to other content types as well, and is presented as such.

The content provider makes available the AdaptationQoS that declares and defines the following IOPins:

- Free IOPins: `PARAM1`, `PARAM2`, …, denoting the set of parameters to be used with a specific adaptation scheme.
- Dependent IOPins: `CODESIZE` (rate) and/or `MSE` (distortion) obtained as a function of the chosen set of parameters (`PARAM1`, `PARAM2`, …)

A UCD may then request minimization of `MSE` subject to `CODESIZE` ≤ *average transmission rate supported by network* times *maximum tolerable delay*. Alternatively, it can request minimization of `CODESIZE` subject to `MSE` ≤ *maximum acceptable distortion value*. The ADTE readily finds the appropriate set of parameters in either case by searching in the space of free IOPins: `PARAM1` `PARAM2`, … etc. and passing it to the BAE that implements the scheme.

A common BAE for JPEG is one that uses a single parameter (`PARAM1`) representing a quality factor (for instance as suggested by the Independent JPEG Group) to use for generating an 8x8 quantization matrix to requantize DCT coefficients. An improvement yielding better quality and lower rate was recently reported [2]. The ADTE returns the right quality factor to use for either UCD type, by searching the space of available quality factors and the corresponding pre-computed rate and distortion achieved by choosing them.

Interestingly, not knowing the actual rate achieved for a given set of parameter(s) is a universal problem in a wide range of reported rate-distortion optimal adaptation methods for various content-types that depend on a Lagrangian parameter λ. The AdaptationQoS in these cases can provide λ as a function of the actual rate achieved to enable deciding the right parameter λ based on a given rate constraint.

B. JPEG2000 image adaptation

A JPEG2000 image contains simultaneous spatial, SNR and component (color) scalability. It may also have precinct-based scalability, but in this example we focus only on the first three. Such a bit-stream can be represented in a 3-dim logical hypercube structure with multiple layers along each of the three scalability dimensions. Useful adaptations of the bit-stream truncate the logical hypercube at the outer ends.

The task of the ADTE is essentially to decide on the number of spatial, SNR and component layers to include based on available constraints. A BAE would use the decisions to actually drop the layers that are not required, and also to conduct other minor update operations on the bit-stream to guarantee syntax conformance.

The content provider makes available the AdaptationQoS metadata that defines and declares the following IOPins:

- Free IOPins: `NSPATIAL`, `NSNR`, `NCOMP` – indicating number of spatial, SNR and component layers respectively.
- Dependent IOPins: `CODESIZE` (rate), `MSE` (computed by reconstruction at the highest resolution for both grayscale and color), `IMAGE_WIDTH`, `IMAGE_HEIGHT` and `ISCOLOR` (whether image is color or grayscale), each as a function of the free IOPins.

Based on this metadata, it is possible to flexibly derive adapted versions based on various considerations represented in the UCD. We show two possible UCDs below that entertain very different considerations for adaptation of an image.

The first UCD requests maximization of `IMAGE_WIDTH` (which also maximizes `IMAGE_HEIGHT`) subject to the

(a) (b) (c)

Fig. 4 JPEG2000 image adaptation. (a) Original image, size 512x515. (b) Adapted with first UCD – max resolution 300x300, max codesize 6000 bytes, display grayscale. (c) Adapted with second UCD – min resolution 100x100, max codesize 6000, display color.

following limit constraints: `IMAGE_WIDTH` ≤ *display width* provided; `IMAGE_HEIGHT` ≤ *display height* provided; `ISCOLOR` cannot be true unless the viewing *display is color capable*; `CODESIZE` ≤ *average transmission rate supported by network* times *maximum tolerable delay* provided. Note the preference here is for a large image, even if that means sacrificing color for a color display or sacrificing quality.

The second UCD requests minimization of `MSE` subject to the following limit constraints: `IMAGE_WIDTH` ≥ *minimum desired display width*; `IMAGE_HEIGHT` ≥ *minimum desired display height*; `ISCOLOR` matches the *color capability of the display* exactly; and `CODESIZE` ≤ *transmission rate supported by network* times *maximum tolerable delay* provided. Note that here the preference is for a better image quality, as long as the size is greater than a minimum desired.

In both cases, the ADTE obtains appropriate solutions by searching the discrete space of free IOPins. But the AdaptationQoS description is agnostic of the considerations used to drive the adaptations. Fig. 4 shows an original image and two adapted versions generated by the above two UCDs with the same codesize constraint. The first adapted version in Fig. 4(b) is grayscale and is at half the resolution of the original. The second adapted version in Fig. 4(c) is color and has better quality, but is quarter the original resolution.

C. Motion compensated predictive video adaptation

Here we discuss the rate adaptation of motion compensated predictive coded video like MPEG-X and H.26X. For such non-scalable bit-streams, a variety of rate-adaptation options exist with varying complexities, such as dynamic rate shaping [3], requantization [4], and object-based transcoding [6].

In [8] a utility-based rate adaptation framework was proposed, which can be illustrated by conducting frame dropping (FD) and coefficient dropping (CD) for adaptation of MPEG-4 video. FD adapts the source stream by skipping frames, while CD operates by truncating some high frequency DCT coefficients. The combination of FD and CD is attractive due to its simple implementation and flexible tradeoff between spatial and temporal qualities.

In order to enable prompt decision-taking subject to constraints, rate-distortion (R-D) information per group of pictures (GOP) is collected by computing sampled FD-CD operations, and transmitted as AdaptationQoS description to the ADTE. The ADTE uses this to obtain the optimal FD-CD decisions per GOP based on constraints, while the BAE uses the decisions to reshape the bit-stream by conducting FD-CD operations. User preferences are also used in decision-taking to yield a valuable extension to traditional R-D optimization.

The content provider makes available the AdaptationQoS description that defines and declares the following IOPins:

- Adaptation unit IOPin: GOP.
- Free IOPins: `NBFRAME`, `NPFRAME`, `RCOEFF` – indicating respectively the number of P-frames to be dropped in a GOP, the number of B-frames to be dropped in a sub-GOP, and the ratio of bit reduction by CD $\in \{0.0,0.1,\ldots\}$, for a given adaptation unit which could be one or more GOPs.
- Dependent IOPins: `BANDWIDTH` (available bandwidth), `UTILIY` (quality of adapted video). In this example PSNR is adopted for Utility. Other quality measurements like mean opinion scale (`MOS`) may also be used.

The following are two typical kinds of optimizations in the FD-CD adaptation, which may be guided by the above AdaptationQoS metadata. First, for each adaptation unit, find optimal operation of FD-CD that maximizes `UTILITY` (`PSNR`), subject to `BANDWIDTH`≤available bandwidth (resource-constrained utility maximization). Second, for each adaptation unit, find optimal operation of FD-CD that minimizes `BANDWIDTH` (bit-rate), subject to `PSNR` ≥ *minimum acceptable quality set by user* (e.g., 32 dB).

Adaptation of 1.5Mbps MPEG-4 video (CIF format with 30fps, GOP size=15, and sub-GOP size=3) down to about 200kbps under dynamic bandwidth constraints was demonstrated in [8]. There is little computational overhead for adaptation decision-taking. Real time FD-CD adaptation in BAE was shown to be feasible on moderate PC systems [9]. Even if the R-D information is unavailable (e.g., for live videos), a content-based utility function prediction approach [9] can be conducted so that the real time decision-taking can still be guaranteed

D. Fully scalable video adaptation

Fully scalable video bit-streams have already been introduced in Section III (see Fig. 2). The ADTE in this case, takes decisions on the number of temporal, spatial and SNR layers to include for each successive GOF (adaptation unit), based on current network and terminal constraints. For streaming sessions, the ADTE should be designed to take decisions for successive GOFs (adaptation units) synchronously with the transmission schedule, in order to accommodate dynamically varying network and usage conditions. Thus, the UCDs and UEDs that actually provide the constraints may change dynamically during a streaming session, causing the ADTE decisions for the currently processed and transmitted GOFs to also change accordingly.

The content provider makes available the AdaptationQoS description that defines and declares the following IOPins:

- Adaptation unit IOPin: `GOF`.
- Free IOPins: `NTEMP`, `NSPATIAL`, `NSNR` – indicating number of temporal, spatial, and SNR layers respectively.
- Dependent IOPins: `FRAMERATE` (temporal resolution), `BITRATE` (rate), `PQUAL` (perceptual GOF quality combining frame SNR computed at the highest resolution with framerate), `FRAMEWIDTH`, `FRAMEHEIGHT` – each as a function of free IOPins per GOF.

The following example UCD requests an adaptation. For the first GOF (adaptation unit), the UCD requests maximization of `PQUAL`, subject to: `FRAMEWIDTH` ≤ *display width* provided; `FRAMEHEIGHT` ≤ *display height* provided; FRAMERATE ≥ *a minimum desired value*; and `BITRATE` ≤ *average transmission rate supported by network*. For all subsequent GOFs, the `FRAMEWIDTH` and `FRAMEHEIGHT` limit constraints are replaced by one that requires them to remain the same as the previous adaptation unit. Thus, the ADTE chooses the spatial resolution only for the first GOF, and maintains the same for all subsequent GOFs. However, the temporal and SNR layers chosen keep changing depending on the video characteristics as provided in the AdaptationQoS and the current network conditions.

We demonstrate the adaptation performance based on this AdaptationQoS and UCD on 288 frames of the CIF *Foreman* sequence, compressed using the MC-EZBC [17] inter-frame scalable video codec. The bit-stream consists of 18 16-frame GOFs, each with 5 temporal, 6 spatial and 5 SNR layers. The parameters in the UCD generate a QCIF resolution adapted video for the first GOF, which is maintained for all subsequent GOFs. We consider two cases: first where the average available transmission rate for the network is 700 Kb/s for the duration of the transmission, and the second where the constraints are dynamically updated every one-third of the video so that for the same average rate of 700 Kb/s, the available rates for each individual one-third are 700 Kb/s, 350 Kb/s and 1050 Kb/s respectively. Table 1 presents for both cases the actual bandwidth transmitted along with the number of temporal, spatial and SNR layers transmitted for each GOF. Adaptation of 288 frames on a moderate PC in both cases took less than 0.5 s, which is much less than that required for 30 frames/s transmission and playback. The results presented use an ad hoc measure of perceptual quality `PQUAL`. A comprehensive study on perceptual characteristics of temporal resolution and frame SNR is needed, in order to enable appropriate decision-taking for fully scalable video.

Table 1. Dynamic adaptation to match available bandwidth. T/S/Q stands for Temporal/Spatial/SNR(Quality) layers preserved by adaptation. All BWs are in Kb/s.

GOF	Constant BW			Dynamic BW		
	Av. BW	Actual BW	T/S/Q Layers	Av. BW	Actual BW	T/S/Q Layers
0	700	600	5/5/2	700	600	5/5/2
1	700	671	4/5/3	700	671	4/5/3
2	700	536	5/5/2	700	536	5/5/2
3	700	544	5/5/2	700	544	5/5/2
4	700	542	5/5/2	700	542	5/5/2
5	700	670	5/5/2	700	670	5/5/2
6	700	657	5/5/3	350	332	4/5/2
7	700	679	3/5/4	350	273	5/5/1
8	700	633	5/5/2	350	321	4/5/1
9	700	669	5/5/2	350	317	4/5/1
10	700	579	5/5/2	350	290	4/5/1
11	700	651	5/5/2	350	308	4/5/1
12	700	687	4/5/3	1050	889	5/5/3
13	700	521	5/5/2	1050	870	5/5/3
14	700	607	4/5/3	1050	978	4/5/4
15	700	665	5/5/3	1050	876	4/5/4
16	700	587	5/5/3	1050	827	4/5/4
17	700	553	5/5/3	1050	1010	4/5/4

VI. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper we have presented the concepts behind the decision-taking framework supported in MPEG-21 (Part 7) DIA to enable Terminal and Network QoS. Casting the adaptation decision-taking problem as a generic constrained optimization problem involving adaptation variables enables creation of universal decision-taking engines that take decisions based on terminal, network and preference constraints, irrespective of their semantics and content type.

Decision-taking by exhaustive search over discrete variables is straightforward and covers the vast majority of practical adaptation scenarios existing today – some of which are presented in the paper. Viable strategies when one or more variables are continuous are also presented. Recent work on MADS seem to be very promising for handling continuous variable ADTE problems with black box limit and optimization constraints. Penalty methods requiring UCD analysis to generate a penalty term may also be explored. We hope the initial directions presented here would lead to more thorough future work and exploration of scenarios where continuous variable adaptation decisions would be useful.

REFERENCES

[1] O. Werner, “Requantization for transcoding of MPEG-2 frames,” *IEEE Trans. Image Processing*, vol. 8, no. 2, pp. 179-91, Feb 1999.

[2] H. H. Bauschke, C. H. Hamilton, M. S. Macklem, J. S. McMichael, and N. R. Swart, “Recompression of JPEG images by requantization”, IEEE Trans. Image Proc., vol. 12, no. 7, pp. 843-49, July 2003.

[3] A. Eleftheriadis and D. Anastassiou, "Constrained and General Dynamic Rate Shaping of Compressed Digital Video", *Proceedings, 2nd IEEE International Conference on Image Processing*, Washington, DC, October 1995, pp. III.396-399.

[4] P. Assuncano and M. Ghanbari, “Post-processing of MPEG2 coded video for transmission at lower bit rates,” in *Proc. IEEE Int. Conf. On Acoustic, Speech, and Signal Processing’96*, vol. 4, Atlanta, GA, May 1996, pp. 1998-2001.

[5] P. Yin, A. Vetro, B. Liu, H. Sun, “Drift compensation for reduced spatial resolution transcoding,” *IEEE Trans. Circ. And Systems for Video Technology*, vol. 12, no. 11, pp. 1009-20, Nov 2002.

[6] A. Vetro, H. Sun, and Y, Wang, “Object-based transcoding for adaptive video content delivery,” *IEEE Trans. Circuit Syst. Video Technol.*, vol. 11, pp. 387-401, Mar., 2001.

[7] Mihaela van der Schaar, Y. Andreopoulos, “Rate-Distortion-Complexity driven Proactive Ubiquitous Multimedia Access,” *IEEE Trans. Multimedia*, *Special Issue on MPEG-21*, in preparation.

[8] J.-G. Kim, Y. Wang, and S.-F. Chang, "Content-adaptive utility-based video adaptation," in *IEEE Int. Conf. on Multimedia and Expo*, Baltimore, Maryland, July 2003.

[9] Y. Wang, J.-G. Kim, and S.-F. Chang, "Content-based utility function prediction for real-time MPEG-4 video transcoding," in *Proc. IEEE Int. Conf. Image Proessing*, Barcelona, Spain, Sep. 2003.

[10] "ISO/IEC 21000-7 FDIS Part 7: Digital Item Adaptation," *ISO/IEC JTC 1/SC 29/WG 11/N6168*, Dec 2003.

[11] A. Vetro "Overview of MPEG-21 Digital Item Adaptation", *IEEE Trans. Multimedia*, *Special Issue on MPEG-21*, in preparation.

[12] "Information Technology – Multimedia Framework (MPEG-21) – Part 1: Vision, Technologies and Strategy," ISO/IEC 21000-1," *2002*.

[13] J. Bormans, J. Gelissen, A. Perkis, *MPEG-21: The 21st century multimedia framework*, Signal Processing Magazine, IEEE, Volume 20, Issue 2, pp. 53 – 62, March 2003.

[14] D.S. Taubman and M.W. Marcellin, "JPEG2000: Image Compression Fundamentals, Standards and Practice," *Kluwer Acad. Pubs, 2002*.

[15] J. –R. Ohm, "Registered Responses to the CfP on Scalable Video Coding," ISO/IEC JTC1/SC29/WG11 MPEG04/M10569, Munich, March 2004.

[16] J. –R. Ohm, M. van der Schaar, J. W. Woods, "Interframe wavelet coding – motion picture representation for universal scalablity," submitted to *Signal Processing: Image Communication*.

[17] S. –T. Hsiang, J. W. Woods, "Embedded video coding using invertible motion compensated 3-D subband/wavelet filter bank," Signal Processing: Image Communications, vol. 16, pp. 705-24, May 2001.

[18] D. Mukherjee and A. Said, "Structured Scalable Meta-formats (SSM) for Digital Item Adaptation," *Proc. SPIE, Internet Imaging IV*, Santa Clara, California, vol. 5018, pp. 148-67, Jan 2003.

[19] D. Mukherjee, Peisong Chen, Shih-ta Hsiang, John Woods and A. Said, "Fully Scalable Video Transmission using the SSM Adaptation Framework," *Proc. SPIE, Visual Communications and Image Processing*, Lugano, Switzerland, vol. 5150, July 2003.

[20] G. Panis, et al., "Bitstream Syntax description: A tool for multimedia resource adaptation within MPEG-21," *Signal Processing: Image Communication (Special Issue on Multimedia Adaptation)*, vol. 18, pp. 597-767, Sep 2003.

[21] "Bitstream Syntax Description," *IEEE Trans. Multimedia*, *Special Issue on MPEG-21*, in preparation.

[22] Ralph E. Steuer, *Multiple criteria optimization: theory, computation and application*, Krieger Publishing Company, 1996.

[23] S. Lerouge, P. Lambert, R. Van de Walle, "Multi-criteria Optimization for Scalable Bitstreams," *Visual Content Processing and Representation*, *8th International Workshop VLBV 2003, Lecture Notes in Computer Science*, vol. 2849, Sept. 2003.

[24] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*, Academic Press.

[25] T. G. Kolda, R. M. Lewis, V. Torczon, "Optimization by Direct Search," New Perspectives on some classical and modern methods," *SIAM Review*, vol. 45, pp. 385-482, Aug 2003.

[26] J. A. Nelder and R. Mead, "A simplex method for function minimization," *Comput. J.*, vol. 7, pp. 308-13, 1965.

[27] V. Torczon, "On the convergence of the multidirectional search algorithm," *SIAM J. Optimization*, vol. 1, no. 1, pp. 123-45, 1991.

[28] R. Hooke, T. A. Jeeves, "Direct search solution of numerical and statistical problems," *J. Assoc. Comput. Mach.*, vol. 8, pp. 212-219, 1961.

[29] V. Torczon, "On the convergence of pattern search algorithms," *SIAM J. Optimization*, vol. 7, no. 1, pp. 1-25, Feb 1997.

[30] C. Audet, J. E. Dennis, "Analysis of Generalized Pattern Searches," *SIAM J. Optimization*, vol. 13, no. 3, pp. 889-903, Feb 2003.

[31] R. M. Lewis, V. Torczon, "Pattern search methods for linearly constrained minimization," *SIAM J. Optimization*, vol. 10, no. 3, pp. 917-41, May 2000.

[32] C. Audet, J. E. Dennis, "Mesh Adaptive Direct Search algorithms for constrained optimization," *Technical Report TR 04-02*, Department of Computational and Applied Mathematics, Rice University, Houston, TX 2004 – submitted to *SIAM J. Optimization*.

[33] C. Audet, J. E. Dennis, "Pattern search algorithms for mixed variable programming," *SIAM J. Optimization*, vol. 11, no. 3, pp. 573-94, Nov 2000.

[34] M. A. Abramson, "Pattern search algorithms for mixed variable general constrained optimization problems," *Ph.D. Thesis*, Dept. of Computational and Applied Mathematics, Rice University, Houston, TX, 2002.

Debargha Mukherjee (M'99) was born in Calcutta, India. He received the B. Tech. Degree in E&ECE from the Indian Institute of Technology, Kharagpur, in 1993; and the M.S. and PhD. degrees in ECE from the University of California, Santa Barbara, in 1995 and 1999 respectively. He is currently a Senior Member of Research Staff, in the Imaging Technology Department of HP Laboratories, Palo Alto, California.

In 1998, he received the IEEE Student Paper Award at the IEEE International Conference on Image Processing, Chicago, Illinois. His research interests include multimedia compression and networking, signal processing, and information theory.

Eric Delfosse received the M.S. degree of Civil Electrotechnical Engineer in Telecommunications from the Free University of Brussels (VUB), Belgium, in 1999. From 1999 till 2001, he has been a Researcher at the VUB's Telecommunication Research Group. In 2001, he joined the Multimedia Image Compression Systems (MICS) group of IMEC research center, Leuven, Belgium. His current research interests include wavelet-based image coding algorithms, 3D Graphics and Quality of Service.

Since December 2001, he is one of IMEC's representatives at the ISO/IEC MPEG standardization committee, where he contributed to the MPEG-4 Animated Framework eXtension (ISO/IEC 14496-16) and MPEG-21 Digital Item Adaptation (ISO/IEC 21000-7) standardization activities.

Jae-gon Kim received the B.S. degree in electronics engineering from Kyungpook National University, Korea, in 1990, the M.S. degree in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Korea, in 1992. Since 1992, he has been a Senior Member of Research Staff, in the Broadcasting Media Research Group of Electronics and Telecommunications Research Institute (ETRI), Korea. He is currently the Team Leader of the Broadcasting Content Research Team. From 2001 to 2002, he was a staff associate at Columbia University, New York. His research interests include multimedia content analysis, representation, compression and networking.

Yong Wang received his B.S. and M.S. degrees from Tsinghua University, Beijing, China in 1999 and 2001 respectively, both in electrical engineering. He is currently a Ph.D. candidate supervised by Professor Shih-Fu Chang in the Department of Electrical Engineering at Columbia University, New York, NY.

From 2001, he has been as a research assistant in the digital video and multimedia (DVMM) group at Columbia University. During the summer of 2003, he was an intern working at HP labs, Palo Alto, CA. From 1999 to 2001, he was a visiting student at Microsoft Research Asia, Beijing, China. His research interests include video coding, adaptation, communication and content-based analysis.